

Jena Research Papers in Business and Economics

Optimally locating in-house logistics areas to facilitate JIT-supply of mixed-model assembly lines

Simon Emde, Nils Boysen

14/2010

Jenaer Schriften zur Wirtschaftswissenschaft

**Working and Discussion Paper Series
School of Economics and Business Administration
Friedrich-Schiller-University Jena**

ISSN 1864-3108

Publisher:

Wirtschaftswissenschaftliche Fakultät
Friedrich-Schiller-Universität Jena
Carl-Zeiß-Str. 3, D-07743 Jena
www.jbe.uni-jena.de

Editor:

Prof. Dr. Hans-Walter Lorenz
h.w.lorenz@wiwi.uni-jena.de
Prof. Dr. Armin Scholl
armin.scholl@wiwi.uni-jena.de

www.jbe.uni-jena.de

Optimally locating in-house logistics areas to facilitate JIT-supply of mixed-model assembly lines

Simon Emde, Nils Boysen

*Friedrich-Schiller-Universität Jena, Lehrstuhl für Operations Management,
Carl-Zeiß-Straße 3, D-07743 Jena, Germany,
{simon.emde,nils.boysen}@uni-jena.de*

Abstract

In modern-day production systems, ever-rising product variety poses a great challenge for the internal logistics systems used to feed mixed-model assembly lines with the required parts. As an answer to this challenge many manufacturers especially from automobile industries have identified the *supermarket*-concept as a promising part feeding strategy to enable flexible small-lot deliveries at low cost. In this context, supermarkets are decentralized in-house logistics areas in the direct vicinity of the final assembly line, which serve as intermediary stores for parts. Small tow trains are loaded with material in a supermarket and deliver parts Just-in-Time to the stations laying on their fixed route. This paper discusses the general pros and cons of the supermarket-concept and treats the decision problem of determining the optimal number and placement of supermarkets on the shop floor. A mathematical model is proposed, an exact dynamic programming algorithm presented, and the validity of the proposed approach for practical purposes is investigated in a comprehensive computational study.

Keywords: Mixed-model assembly lines; Just-in-Time; Material supply; Tow Trains

1 Introduction

High-volume production systems have traditionally been implemented as flow lines to benefit from economies of scale and productivity gains. Modern mixed-model assembly lines are capable of producing a great number of variants of a common base product on the same line with negligible set-up times and cost (lot size one). In recent years,

an undeniable trend towards increasing product variety, in order to satisfy diversified customer demand, can be noted (see, e.g., Boysen et al., 2009b). Consequently, today’s mixed-model assembly systems have an enormous hunger for a very diverse range of parts, making the organization of a well-run logistics network one of the most vital tasks to ensure that final assembly runs smoothly and efficiently.

One significant challenge within this context is the feeding of parts to the productive units (stations) at the line. On the one hand, material and parts must always reach the work stations on time to avoid extremely costly line stoppages. On the other hand, excessive stock at the stations and/or shop floor traffic lead to high handling and holding cost. Following the just-in-time (JIT) principle, a mounting number of manufacturers is therefore adopting the so-called *supermarket*-concept. In this sense, supermarkets are decentralized storage areas scattered throughout the shop floor which serve as an intermediate store for parts required by nearby line segments. From these stores, tow trains (or tuggers) – towing vehicles connected to a handful of waggons – set off according to a fixed schedule, delivering parts from the supermarket to, and collecting empty bins from, the stations laying on their pre-determined routes. Finally, empty trains return to the supermarket and are refilled for their next tour. This way, decentralized supermarkets enable frequent small-lot deliveries of parts, so that inventory at the line is reduced and long-distance deliveries from a central receiving store are avoided.

One important optimization problem in this regard is the determination of the number and placement of the supermarket areas. Shop floor space is very valuable, creating too great a number of supermarkets will therefore often entail more cost than benefits. Having too few, poorly placed supermarkets, on the other hand, will greatly diminish their positive effects, making them, in the worst case, barely better than traditional centralized stores. Using the available space as well as possible by selecting the optimal number of supermarkets and strategically locating them is therefore critical. This paper presents a polynomial-time dynamic programming scheme to optimally solve this problem.

The remainder of this paper is organized as follows. Section 2 will discuss the operation and benefits of a JIT-supermarket in more detail and review the extant literature. Section 3 will present a formal description of the supermarket location problem as well as a suitable mathematical representation. In Section 4, we will introduce the optimization procedure and in Section 5 we will investigate the practical benefits of choosing the optimal placement and number of supermarkets in a comprehensive computational study. Finally, Section 6 concludes the paper.

2 Operation of a JIT-supermarket and literature review

Most OEMs in the automotive sector today employ – or at least aim to employ – JIT strategies to keep final assembly well supplied while also minimizing work-in-process. Due to the enormous product variety and consequent part diversification, efficient in-house logistics are especially crucial to secure competitiveness as storage space at the stations is usually the most scarce and the most expensive.

Traditionally, stations have been served from a single central storage area from which

parts were brought pallet- or binwise to the line in individual deliveries. Such a centralized part supply is not very much in line with the JIT-principle. First, as stations are supplied individually and from a potentially far away central store, parts have to be delivered in moderate to large lots because otherwise shop-floor traffic would be unmanageable. This entails an increase in in-process inventory and moves reorder dates forward. Second, the pallets, once delivered, will have to be kept at the stations, where space is notoriously scarce (see Boysen et al., 2009a). This may seriously hamper workers and reduce productivity. Finally, from time to time unforeseen events, e.g., material defects, occur requiring emergency deliveries of missing parts to avoid line stoppage. Clearly, an express delivery from far distant central store is very time consuming. As a reaction of these disadvantages some manufacturers started using consignment warehouses (e.g., Valentinia and Zavanella, 2003, Boysen et al., 2008) operated by a third party logistics provider. Whenever a station runs out of parts, a pallet of new ones is commissioned to replace them. However, consignment stocks merely transfer the problems to a third party, which is not in line with the basic principles of cooperative supply chain management. At least over the long run higher logistics cost will be passed on to OEMs via the consignment contract.

To alleviate these problems, supermarkets, that is, decentralized logistics areas where parts are stocked for close-by line segments, were introduced. As these facilities substitute sporadic large-lot for frequent small-lot deliveries, supermarkets can be seen as the in-house equivalent of cross-docks (see, e.g., Apte and Viswanathan, 2000, Boysen and Flidner, 2010). Parts are usually brought to these sites by relatively large industrial trucks. The materials are then sorted and intermediately stored to be loaded onto tow trains and delivered to the stations just as needed. To facilitate a dependable part supply, parts are typically transported in standard-size bins of constant capacity. Also, to keep congestion to a minimum and improve reliability, tow trains are normally operated on a fixed schedule, which determines exactly the time when the tugger leaves the supermarket and of each stopover on its ordained route. Some automobile producers have even automated these stopovers by employing so-called “shooter-racks” (Emde et al., 2009). This special kind of gravity flow rack allows the tugger to dock, opening gates at the back of the rack and the docked waggon such that full bins are ejected onto the rack by elastic springs while empty ones slide back to the train. These shelves reduce the unloading time to just a few seconds and render reliable schedules possible. Moreover, one automobile producer we visited was experimenting with display panels installed at each station similar to those of bus and railway stations. Here, a countdown until the tugger’s next arrival was announced, so that anticipating material shortages in a credible and timely manner got much easier for assembly workers and team leaders.

“Kanban supermarkets” are a part of the famous Toyota Production System (Vatalaro and Taylor, 2005, Holweg, 2007) and have a long tradition in many industries (Rees et al., 1989, Hodgson and Wang, 1991, Spencer, 1995). However, in-house logistics of this kind were often organized as pure kanban systems. Seeing that production sequences are usually known well in advance (about three to four days in the automotive industry), the bill of materials exactly determines how many parts are necessary within a given time

interval at any station. Classic kanban systems as envisioned by the Toyota Production System (Monden, 1998) do not exploit this information, because they only refill parts that have already reached a critical stock level. However, waiting to be “surprised” by low stocks, which might necessitate emergency deliveries if the next scheduled stopover at the station is too far off, is needless when deterministic information about the production schedule and hence part demand as well as delivery schedules is available. As a consequence, modern, well-planned supermarket-based feeding systems can run more smoothly and with less manpower and fewer vehicles than many purely kanban-based systems (Golz et al., 2010).

Moreover, the advantages of decentralized logistics networks have been widely discussed in the literature (e.g., Johnson and Leenders, 2004, Wanke and Zinn, 2004) and apply just as well to in-house logistics: Shorter delivery times by being closer to the consumer (i.e., the assembly line), freight consolidation by being supplied by industrial trucks, and faster turnover by stocking and delivering parts just as needed. Furthermore, apart from the obvious benefits of being more in line with the original goals of the JIT philosophy, frequent small-lot deliveries as enabled by supermarkets can also be easily replanned while large-lot deliveries, once made, are hard to revoke, which is an important advantage in the event of unforeseen disturbances. Also, comparatively small bins can be stored in easily accessible racks close to the line, where workers can take parts in an ergonomic and efficient manner, which reduces handling times and the strain on the workforce.

On the downside, supermarkets consume space on the factory floor, which is scant and expensive. Parts are stored in shelves designed for ease of access such that workers can prepackage parts in a comfortable manner, analogously to customers in a traditional supermarket, and thus they are typically less space-efficient than traditional warehouses. An effective implementation of the supermarket-concept will also necessitate some investment in equipment, staff and maintenance. Finding the optimal compromise regarding this trade-off and investigating the operational benefits of supermarkets are important issues which this paper will tackle in the following.

The planning and control of this in-house logistics concept amounts to a complex task involving several interrelated decision problems:

- (i) Decide on the number and location of decentralized supermarkets.
- (ii) Determine the number of tow trains per supermarket and assign line segments to them.
- (iii) Determine each tow train’s fixed delivery schedule.
- (iv) Decide on the bins to be loaded per tour of a tow train.

To date, there are only a few publications explicitly dealing with any of the above problems. Emde et al. (2009) tackle the loading of tow trains given their routes and schedules and develop a polynomial time exact algorithm to minimize the number of bins stored at the line given a limited tugger capacity. Vaidyanathan et al. (1999) formulate problem (ii) as a capacitated vehicle routing problem where stations have a constant

parts consumption rate over time. The authors use a nearest neighbor heuristic to generate start solutions and subsequently the 3-opt heuristic to improve these solutions. Emde and Boysen (2010) solve problems (ii) and (iii) jointly with an optimal nested dynamic programming scheme, with the goal of minimizing an aggregate of the stock at the stations and the number of vehicles required to keep them supplied, which runs in polynomial time. Golz et al. (2010) examine a real-world implementation of the supermarket concept at a major German motor company and develop a heuristic procedure to decide on the routes (ii), the schedules (iii) and the load (iv) of the tow trains, given a single supermarket and a set of possible routes to choose from, aiming to minimize the number of vehicles and operators while avoiding stock-outs at the line. Battini et al. (2010) consider the problem of locating the optimal number of supermarkets on the factory floor, i.e., problem (i). The authors contemplate an assembly system with multiple parallel production lines where the entire lines (not individual stations of a single line) have to be supplied with parts. In a first step, they group the lines together according to their degree of component commonality, and then, in a second step, they assign supermarkets to locations such that the distance from each supermarket to the group of lines assigned to it is minimal. However, it was our observation in the automobile industry that typically multiple supermarkets are applied per line. While, to the best of the authors' knowledge, the aforementioned article is the only paper explicitly dealing with problem (i), the issue of placing in-house logistics areas bears some resemblance to classic facility location problems (for a survey, see, e.g., Klose and Drexl, 2005). These models are only partially applicable to the supermarket location problem, however, as they ignore the specifics of assembly production systems, especially the layout of the assembly line on the factory floor. One side-effect of this is that practically all extant multi-facility location problems are NP-hard. As we will see, this is not the case for the present problem. Also, facility location models usually only seek to minimize the sum or the maximum of distances between facility and customers. We will discuss in the next section why this is an inadequate goal in our case.

3 Problem description

We are assuming a classic mixed-model assembly line, along which there are $s = 1, \dots, S$ productive units (stations), which each have to be supplied with d_s bins of parts from a supermarket. Each station's position on the factory floor is identified by a two-dimensional coordinate (a_s, b_s) , which defines the parking position of a tugger for supplying station s , e.g., at the position of the shooter rack. With these coordinates, distances e_s between station s and $s + 1$ as well as distances z_{is} between station s and supermarket i , $i = 1, \dots, n$, to be covered by a tugger when visiting these locations, can easily be calculated according to some metric. The Supermarket Location Problem (SLP) is then defined as follows: What is the optimal number n of supermarkets and which stations should each supermarket serve? Note that, once the latter question is answered, calculating the positions of the supermarkets is assumed to be trivial since we presuppose a location problem in the plane.

Figure 1 conceptually shows the layout of a factory floor with two decentralized logistics areas. The central receiving store distributes parts in large-lot deliveries to the supermarkets, where they are prepackaged and sorted to be loaded as needed on the tow trains, which circulate through the stations in the respective supermarket's supply area. Note that, in a subsequent step, the process of actually delivering parts to the stations necessitates the routing, scheduling and loading of the tow trains assigned to the supermarket. While all these problems are certainly interconnected, it is difficult to see how they can be solved concurrently in practice since the placement of a supermarket is a decision with a long lead time, while routes and especially schedules and loads can only be sensibly assigned once information about the production sequence is available, which is normally only a few days in advance. This paper will therefore concentrate on the location problem per se and will apply an approximate measure for estimating the impact of a supermarket location on the operational decision problems.

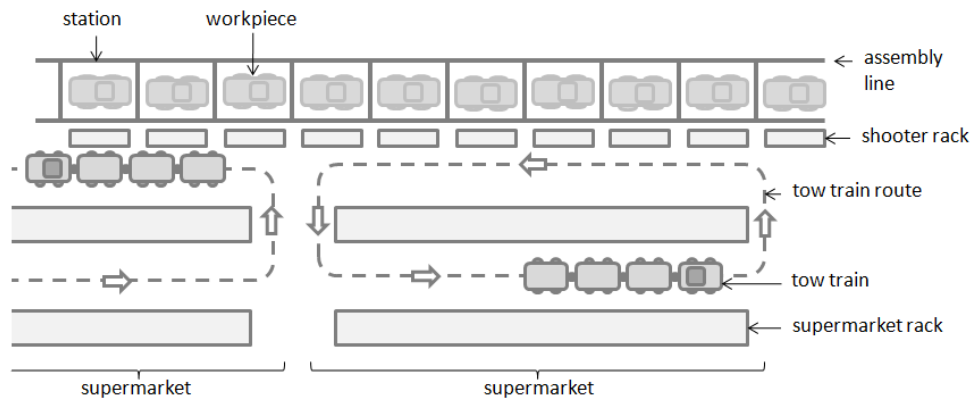


Figure 1: A factory floor concept with two supermarkets.

To accurately model the ensuing facility location problem, we introduce the following premises:

- Stations served by one supermarket are always consecutive, meaning, for instance, that it cannot be that one supermarket delivers parts to stations 1, 3 and 5, and another takes care of stations 2 and 4. Plant managers typically try to avoid interleaving or overlapping supermarket areas as this makes the routing of the vehicles more difficult and leads to inefficiencies in the daily operation of the parts feeding process. For example, we are aware of one case at a major German automobile producer's plant where tow trains often have to wait in front of an automated retractable barrier for the way to be clear before they can continue their rounds. Such unproductive idle times are highly undesirable, of course, and could be avoided by a more advantageous placement of the supermarkets.
- Because bottlenecks can always be allayed (at a cost) by additional routes, vehicles and tighter schedules, no hard restrictions on the number of stations that can

be supplied by one supermarket are necessary. If distances and loads at a single supermarket are great, such setups will be naturally punished in the objective function.

- Parts are packed in bins of identical standardized size (see de Souza et al., 2008, on how this may be done), which is a requirement of the aforementioned shooter racks.
- Routes, schedules and loads for the individual tow trains can only be determined at a later time, when production sequences are known. The SLP must therefore be solved on the basis of aggregated estimates.
- Considering that fast deliveries and short routes are their very purpose, supermarkets are always placed close to the line. Exactly how close depends on the size of the stations themselves as well as the amount of space reserved for driving lanes. We will assume that the minimum offset by which the supermarket must be removed from the line is constant all along the assembly line.
- Since parts are not delivered point-to-point from the supermarket to each station but rather during extensive tours with multiple stopovers, supermarkets need not be close to *all* the stations in their respective area but only to the start and end points of their associated routes. Therefore, supermarkets are placed in the exact middle between the two outer stations in its supply area. Whether or not this spot is actually optimal depends on the way the tuggers will be routed later on, which is unknowable from the point of view of the SLP. Also, the space so chosen may not be available at all if it is already occupied by other facilities and/or equipment. In this case, a different, discrete planning approach would be necessary, which is beyond the scope of this paper. In these instances, the SLP will at least provide a good approximate solution, however.

Concerning the goal of the optimization, the classic minsum-aim of minimizing the sum of the weighted distances from each supermarket to the stations it supplies, often used for location problems in the plane (e.g., Owen and Daskin, 1998, Krarup et al., 2002, ReVelle and Eiselt, 2005), is certainly applicable, if the peculiarities of tow train transportation are provided for. Tuggers will usually not visit each station individually but travel on predetermined routes with multiple stopovers. The path to a station is therefore not a straight line from the supermarket but instead, when calculating distances, we must account for the way tuggers will eventually travel: First, from the supermarket to the first station on its route, then from station to station and, finally, back to the supermarket to refill. Even so, minimizing distances alone does not necessarily guarantee good solutions. Since tuggers have a limited capacity, how many and which stations can be supplied in one tour depends on the parts consumption at those stations. Not too many high-demand stations can be on one route, or else the tow train will not be able to serve them all. Overworked supermarkets will have to make use of additional vehicles, routes and safety stock, all leading to higher operating cost. Distances should therefore be

weighted by the total demand at the stations supplied by each supermarket. As was already mentioned, the exact parts consumption in a given shift will only be known once production sequences have been determined. Regardless, aggregate bin demands per station and shift can usually be estimated with some accuracy, as this does not require intimate knowledge of the exact timing and composition of the model sequence but only an appraisal of average production volumes. Finally, as a third component, fixed cost for creating and maintaining a supermarket should also be accounted for. Taking these points into consideration, using the notation listed in Table 1, we can define the Supermarket Location Problem as follows:

S	number of stations (index $s = 1, \dots, S$)
n	variable encoding the number of supermarkets (index $i = 1, \dots, n$)
Γ	fixed cost per supermarket
e_s	distance from station s to station $s + 1$
d_s	expected number of bins in demand at station s per shift
a_s	x-coordinate of station s
b_s	y-coordinate of station s
x_i	variable encoding the first station supplied by supermarket i
z_{is}	continuous variable encoding the distance from supermarket i to station s

Table 1: Notation

Given $s = 1, \dots, S$ consecutive stations to be supplied with parts from the supermarket, the Supermarket Location Problem consists of partitioning these stations into a variable number of $n = 1, \dots, S$ disjunct subsets, each served by a separate supermarket. A solution is encoded as a vector $X(n) = \{1, x_2, x_3, \dots, x_n, S + 1\} \rightarrow \{2, \dots, S\}$, where x_i marks the left-most station in a supermarket's supply area; consequently, the right-most station served by supermarket i is determined by $x_{i+1} - 1$. Since all stations need to be included in some supermarket's delivery area, the first supermarket's left-hand station is always 1, and the last one's right-hand station is always S . The vector is of length $n + 1$ and its members are to be set such that objective (1) is minimized and constraints (2) and (3) are observed.

(SLP) Minimize $F(X(n)) =$

$$\sum_{i=1}^n \sum_{s=x_i}^{x_{i+1}-1} d_s \cdot \left(\sum_{\tau=x_i}^{x_{i+1}-2} e_{\tau} + z_{i,x_i} + z_{i,x_{i+1}-1} \right) + n \cdot \Gamma \quad (1)$$

subject to

$$x_{i+1} \geq x_i + 1 \quad \forall i = 1, \dots, n \quad (2)$$

$$z_{is} = \left| a_{x_i} + (a_{x_{i+1}-1} - a_{x_i})/2 - a_s \right| + \left| b_{x_i} + (b_{x_{i+1}-1} - b_{x_i})/2 - b_s \right| \quad \forall i = 1, \dots, n; s = 1, \dots, S \quad (3)$$

Objective function (1) aims to minimize the sum of the number of supermarkets n weighted by Γ and the sum over all supermarkets of the total demand at the stations the respective supermarket supplies multiplied by the length of a route through all the stations and back. Note that the calculation of distances mimics the route a tow train setting off from the supermarket will have to travel: First, the distance z_{i,x_i} from supermarket i to the first station x_i in its supply area, then the distance $\sum_{\tau=x_i}^{x_{i+1}-2} e_{\tau}$ from station to station, and finally the distance $z_{i,x_{i+1}-1}$ back to the supermarket. Constraints (2) make sure that there are no overlapping supply areas, and constraints (3) calculate the distance from each supermarket to each station. Considering that factory floors are usually characterized by line-side driving lanes and sharp turns, the Manhattan (or rectilinear) metric is typically best suited to measure distances, although other metrics could also be used. Supermarkets are placed in the middle of the two end-points of its supply area, because tuggers will have to start their tour setting off from the supermarket and finally end it by returning, probably from the other end of the supermarket's supply area. Note that, in many cases, supermarkets can of course not be located exactly in the middle of two stations because the assembly line will probably run through there. However, since supermarkets are, according to the premises, always removed from the line by a constant offset which is identical everywhere, this need not be explicitly modeled.

One practical problem which may arise when using the above objective function is caused by the inclusion of two very different cost factors in one joint objective function: On the one hand, there is the term which estimates future operating cost and aims to minimize the distances and demands in all supermarkets' supply areas, which will tend to be lower the more supermarkets there are. On the other hand, there are capital, staffing and maintenance cost inferred by creating new supermarkets. While a trade-off between these two factors most certainly exists, setting an exact cost coefficient Γ may be difficult in practice. We will therefore propose an algorithm in the next section which computes all non-dominated pairs of operating cost estimate and supermarket count n . All these solutions are optimal for their respective n . An experienced shop floor manager can then easily visualize the inherent trade-off and choose the best set-up for their specific situation (or Γ).

4 Solution procedure

For each supermarket i , the left-most station x_i in its supply area must be determined. This will automatically set the right-most station of the preceding supermarket's area as $x_i - 1$. Since the sets of ordered stations supplied by each supermarket are non-overlapping and distance and demand values depend only on the area of the current supermarket and not on those that come before or after, optimal solutions can be efficiently constructed by a dynamic programming approach.

Let k denote the first station in a supermarket's area, $G(k)$ the minimal cost for the station interval from 1 to $k - 1$ with $G(1) := 0$, $f(j, k)$ the objective value for the supermarket that serves station j through $k - 1$, as determined by

$$f(j, k) = \sum_{s=j}^{k-1} d_s \cdot \left(\sum_{\tau=j}^{k-2} e_\tau + \hat{z}_j + \hat{z}_{k-1} \right),$$

where \hat{z}_s is calculated as with Equation (3) (with $x_i := j$ and $x_{i+1} := k$). The dynamic programming recursion is then defined as

$$G(k) = \min_{1 \leq j \leq k-1} \{G(j) + f(j, k)\}.$$

The goal is now to find the path to $S + 1$ with the lowest $G(S + 1)$ for a given number n of supermarkets. A formal description of the forward-recursive DP-procedure computing the efficient frontier with all non-dominated $(n, p_n, G_n(S + 1))$ -pairs (where n is the total number of supermarkets, p_n encodes the corresponding optimal solution, which can be decoded by a simple backward recovery, and $G_n(S + 1)$ is the objective function value associated with the solution) is shown in Figure 2. Note that solutions thus acquired do not incorporate the fixed cost Γ per supermarket. The optimal solution to program (1)-(3), however, can easily be found by adding $n \cdot \Gamma$ to each $G_n(S + 1)$ in the set of non-dominated solutions the algorithm computes, $\forall n = 1, \dots, S$, and then picking the solution with the lowest total objective value.

```

1 for  $k := 2$  to  $S + 1$  do
2    $p_1(k) := 1$ ;
3    $G_1(k) := f(1, k)$ ;
4 end
5 Store non-dominated solution  $(1, p_1, G_1(S + 1))$ ;
6 for  $n = 2$  to  $S$  do
7    $G_n(k) := \infty \forall k = 1, \dots, S$ ;
8   for  $k := n + 1$  to  $S + 1$  do
9     for  $j := n$  to  $k - 1$  do
10      if  $G_{n-1}(j) + f(j, k) < G_n(k)$  then
11         $p_n(k) := j$ ;
12         $G_n(k) := G_{n-1}(j) + f(j, k)$ ;
13      end
14    end
15  end
16  Store non-dominated solution  $(n, p_n, G_n(S + 1))$ ;
17 end

```

Figure 2: Dynamic programming algorithm for constructing the efficient frontier for the SLP.

Example: Consider the example data given in Table 2. As described in Table 1, d_s denotes the expected demand at each station s in bins of parts for a typical production shift, the horizontal and vertical coordinates of each station are given by a_s and b_s , respectively, and the distances e_s between stations can easily be computed by applying the (in this case) Manhattan metric to these coordinates.

station s	1	2	3	4	5
d_s	3	7	1	8	10
a_s	1	5	12	12	12
b_s	1	1	1	10	17
e_s	4	7	9	7	

Table 2: Example data.

Figure 3 shows the dynamic programming graph for this problem. Nodes stand for stations (with node 6 being the dummy node $S + 1$), while arcs denote the set of stations served by one supermarket. An arc from node 2 to node 5, for example, signifies a supermarket serving stations 2, 3 and 4. The respective arc weight, then, equals $f(2, 5) = (7 + 1 + 8) \cdot (7 + 9 + 8 + 8) = 512$. Grey arcs were added in Lines 1 to 4 of the algorithm in Figure 2 and thus signify paths with only one arc from node 1 to all the other nodes. Building on these grey arcs, the black arcs, added in the first iteration through the for-

loop in Line 6, express paths from the source to all other nodes with exactly two arcs. Note that these will be added to again to get all paths with $n = 3$ arcs etc. Paths that end in the sink node 6 represent complete, feasible solutions to the SLP. Storing the least cost path to 6 in every iteration, i.e., for every arc/supermarket count, yields the efficient frontier, as depicted in Table 3. Adding $\Gamma \cdot n$ to each $G_n(S + 1)$ for every non-dominated solution and selecting the lowest value of these results in the optimal solution to the SLP. The optimal solution for $\Gamma = 300$ in the example is bold in the table.

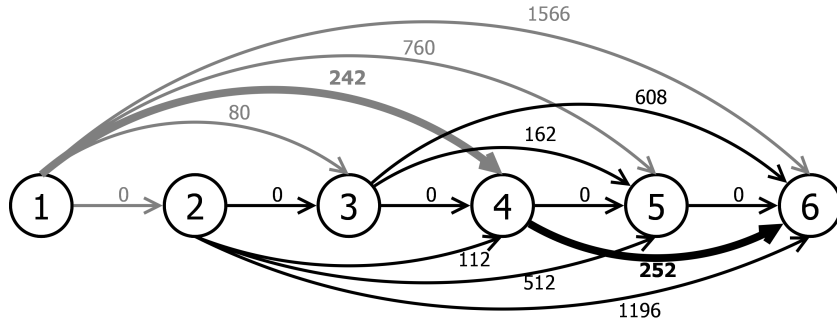


Figure 3: DP-graph in the example. Optimal solution with $n = 2$ edges is bold.

n	1	2	3	4	5
$G_n(S + 1)$	1566	494	242	80	0
$F^*(X(n))$	1866	1094	1142	1280	1500

Table 3: Non-dominated solutions in the example. Optimal solution with minimal $F^*(X(n))$ for $\Gamma = 300$ is bold.

Concerning the time complexity of the proposed algorithm, the DP-graph consists of $S + 1$ nodes, each connected to no more than S others. The consequent maximum of S^2 edges needs to be evaluated up to S times, once for each possible number of supermarkets n . To evaluate an arc, its weight must be calculated which entails the computation of function f , which may involve, in the worst case, summing over S stations. The total number of steps required to optimally solve the SLP is therefore bounded by $O(S^4)$.

5 Computational study

5.1 Instance generation

As there are no established test data for the SLP, we will first describe how the instances used in our computational study were generated. An instance of the SLP is defined by a number of stations S , to each of which a demand d_s and two coordinates a_s and b_s are assigned. The demand d_s is computed by randomly generating forty sequences

of 400 models each. Every model $m \in M$ has a random consumption of part $p \in P$, given as $c_{pm} = \lfloor \text{rnd}(u_m, u_m) \rfloor$, $\forall m \in M, p \in P$, where $u_m = \text{rnd}(0.5, 0.5)$, $\forall m \in M$, $\text{rnd}(\mu, \sigma) \sim N(\mu, \sigma)$ is a normally distributed random number greater than 0 and $\lfloor \cdot \rfloor$ denotes rounding to the nearest integer. Parts come in bins of differing capacity (to reflect the fact that while bins are of standardized size, parts vary in shape and form), namely a uniformly random number from the interval $[1; 100]$, and three different kinds of parts are used at each station. Given these data and a production sequence, bin demand per shift (where a shift is assumed to have $D = 400$ production cycles) and station can easily be calculated (see Emde et al., 2009, for details) and averaged over all forty sequences, i.e., shifts, resulting in the required d_s .

Coordinates a_s were set to $a_s = a_{s-1} + \text{rnd}_u(1, 6)$, $\forall s = 2, \dots, S$, where $\text{rnd}_u(1, 6)$ stands for a randomly drawn number from the interval $[1; 6]$ and $a_1 := 1$. b_s was fixed at 1, $\forall s = 1, \dots, S$, for simplicity's sake. In this computational study, coordinates are relative, meaning that a point $(1, 1)$ is 4 units away from $(5, 1)$. A "unit", in this study, refers to the average distance the tow train can travel in the time span of one production cycle (usually about 60 to 90 seconds in the automotive industry). The parameters used for instance generation are shown in Table 4. For each of the station counts S listed in the table, thirty instances were created, leading to a total of $7 \cdot 30 = 210$ instances, each built on forty sequences of 400 units.

Symbol description		values
S	number of stations	10, 30, 60, 100, 150, 200, 300
$ M $	number of distinct models	100
D	number of production cycles / units per sequence	400

Table 4: Parameters for instance generation

5.2 Computational results

The SLP objective function (1) is, as was already mentioned, merely an estimator of future operating cost. The exact benefits of having the line supplied by a well-placed supermarket will only become obvious once production sequences have been determined and routes and schedules for the tow trains stationed at the supermarket planned. It is conceivable that a location which is thought to be optimal from the point of view of the SLP turns out to be impractical later on due to the difficulty of finding good routes and timetables. To investigate whether or not this is indeed the case, we implemented the proposed algorithm in C# 2008 and ran a series of tests on an x86 PC with an Intel Core 2 Quad Q9550 2.8 GHz CPU and 4096 MB of RAM. We will use the optimal supermarket locations as computed by the dynamic programming procedure presented in this paper as the input parameters for a series of simulation tests. To simulate the day-to-day operation of the supermarkets, we have the assembly line produce the forty sequences

of 400 workpieces each, which on average create the per-station bin demand d_s used in the SLP-optimization, meaning that on average, sequences will obey the anticipated distribution of parts demand, while each individual sequence will most probably deviate from it, as can be expected in practice. Given these deterministic sequences, part and bin demand at each station in each work cycle is known with certainty. Distances from/to the supermarket and between stations are given by the placement in the optimal SLP-solution. In addition to the travel time, there is also a replenishment interval of five cycles that the tugger has to wait out at the supermarket to be reloaded. Each tow train can carry up to fifty bins. The operating cost for each sequence is then measured as the sum of all bins lying in stock over every station and every work cycle, as well as the number of tow trains required to supply the stations in the supermarket's area, the latter weighted with a factor of 1000. The simulation proceeds as follows:

- i. Add a tow train to the (otherwise empty) set of scheduled tow trains T , and set its departure time such that the train arrives at station 1 in work cycle 1.
- ii. If T is empty, go to vii, else pick and remove the tugger t with the earliest departure time c_t from T and have it set off at that time. Set $c_t := c_t + z_1$, where z_1 is the distance from the supermarket to the first station in its supply area. Set $s := 1$.
- iii. When arriving at station s , tow train t unloads as many bins as are required at that station until t 's next scheduled arrival, i.e., the current time c_t plus the duration of a complete tour.
- iv. If, due to capacity or scheduling constraints, tugger t is unable to deliver the required amount of bins, add another tow train to T such that it arrives just-in-time at the undersupplied station to rectify this.
- v. Tugger t will then continue on to the next station ($c_t := c_t + e_s$ and then $s := s + 1$), except if it has already visited all stations (i.e., $s = S$), in which case it will return to the supermarket, be replenished and set off again for station $s := 1$ without delay ($c_t := c_t + z_S + z_1 + 5$).
- vi. Repeat steps iii - v if the current time c_t is still within the planning horizon of 400 work cycles, otherwise go to step ii.
- vii. Identify the maximum number of tuggers en route simultaneously at any one time. This is the number of required vehicles. If a tugger did not actually deliver any bins during its complete tour, do not count it. If bins have been unloaded at a station in a work cycle earlier than that in which they are consumed, they have to be stocked in the meantime. Sum up the number of bins lying in stock at each station in each work cycle.

The basic idea of this procedure is to try to supply all stations with one tow train making cyclic deliveries, i.e., setting off from the supermarket in regular intervals. If that fails due to its limited capacity, another vehicle is scheduled to help out, also making

cyclic deliveries. If that is still not enough, another tugger must be launched and so on. Obviously, assigning routes and schedules in such a way need not necessarily be optimal. However, it is rare to see more sophisticated delivery policies in practice (Emde and Boysen, 2010). If the simulation is repeated for all supermarkets on the factory floor and for all sequences, the average vehicle count and stock level at the stations can be measured, giving a fair appraisal of the operational cost and benefits supermarket-based feeding system.

S	10	30	60	100	150	200	300
ρ	0.89	0.97	0.99	0.99	0.99	1	1

Table 5: Correlation ρ of SLP objective values and actual stock at the stations ($\Gamma = 0$, $n = 5$).

Table 5 lists the Pearson product-moment correlation coefficients ρ for all instances, split on the number of stations S . The table explores the connection between SLP objective F and the actual number of bins stocked at the line as calculated in the simulation experiment, averaged over all forty random sequences per instance. For better comparability, the number of supermarkets was fixed at $n = 5$ and Γ was set to 0, so as to avoid distorting objective values with arbitrary fixed cost. As can be seen, the correlation between SLP objective and actual performance is great throughout, with the lowest correlation coefficient coming to 0.89. All coefficients are significant at the 99% confidence level. Note that, obviously, the exact strength of the correlation will somewhat depend on the weighting factor associated with the vehicle count. However, such a strong correlation as shown in this study suggests, at the very least, that the minsum consideration of distances and demands in the objective function is adequate for predicting future operating success of the supermarkets. Figure 4a visualizes this.

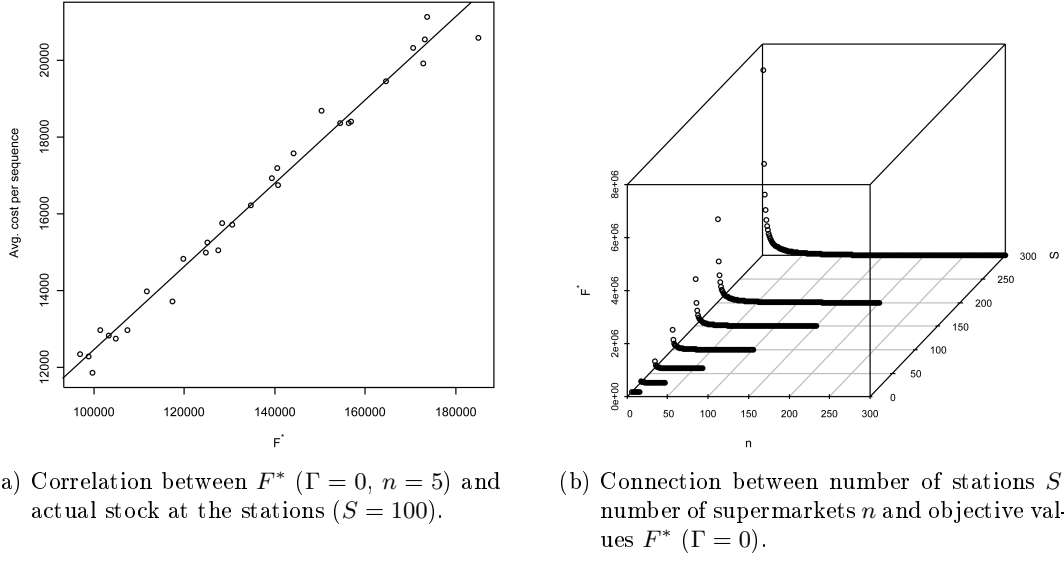


Figure 4: The relation of SLP goodness to actual stock at the line, and to station and supermarket counts.

Apart from the optimal placement of supermarkets, determining the optimal number of them is also a central purpose of the proposed algorithm. While the individual cost associated with operating additional logistics areas, expressed by the parameter Γ in the mathematical program, differs strongly with regard to scarceness of space, available equipment, wage levels and other factors pertinent to specific production settings, some generalizable insights can nonetheless be won. Table 6 contains all F^* values for each possible supermarket count $n = 1, \dots, 10$ with $\Gamma := 0$ and $S = 10$ stations, averaged over the thirty instances. Clearly, the more supermarkets are already installed, the less useful each additional one becomes. F^* values decline very sharply for small n but flatten out quite soon. This is even more obvious in Figure 4b, which depicts the efficient frontiers for all tested station counts. Even in instances with 300 stations, it seems hardly worthwhile to maintain more than three or four supermarkets as afterwards the marginal utility is diminishing quickly.

n	1	2	3	4	5	6	7	8	9	10
F^*	6526.5	2737.4	1463.6	861.3	524.6	309.1	177.0	89.5	34.3	0

Table 6: Average optimal objective function values F^* ($S = 10$, $\Gamma = 0$) for $n = 1, \dots, 10$.

6 Conclusion

This paper deals with an important problem arising in the context of in-house logistics, namely the siting of intermediary stores, “supermarkets”, on the factory floor. To this

end, operational and economic aspects of installing and maintaining supermarkets are discussed, a formal problem description of the Supermarket Location Problem (SLP) is derived, and a dynamic programming scheme to optimally solve the problem is presented. It is shown that the proposed algorithm is suitable for solving instances of realistic size, seeing that the runtime is polynomially bounded and all problems in the extensive computational tests could be solved in under a second of CPU time. It is also evident from comprehensive simulation experiments that solutions deemed “good” by the SLP are in fact effective in reducing operating cost, that is, in-process inventory as well as the number of required vehicles and operators. Moreover, the numeric study strongly suggests that supplying an assembly line from even a small number of supermarkets, say three or four, is vastly superior to a setup with only one centralized storage area. Some challenges for future research still remain, however:

- Like all location problems in the plane, the presented model and algorithm are valid only if supermarkets can be placed anywhere on the factory floor. If, however, only specific areas on the shop floor are available for construction of a supermarket a different, discrete approach is necessary.
- In some assembly systems, other factors than distance and demand may also play into the placement decision, for example, if the same parts are used at multiple stations, it may in some cases be efficient to have these stations served by one supermarket due to consolidation effects.

These challenges seem to be worthwhile fields of research, considering the enormous potential for reduction of costs made possible through well-run supermarket-based parts feeding systems, as shown in this paper.

References

- Apte, U.M.; Viswanathan, S. (2000): Effective cross docking for improving distribution efficiencies. *International Journal of Logistics* 3, 91–302.
- Battini, D; Faccio, M.; Persona, A.; Sgarbossa, F. (2010): “Supermarket warehouses”: stocking policies optimization in an assembly-to-order environment. *International Journal of Advanced Manufacturing Technology* 50, 775–788.
- Boysen, N.; Fliedner, M. (2010): Cross Dock Scheduling: Classification, Literature Review and Research Agenda. *Omega* 38, 413–422.
- Boysen, N.; Fliedner, M.; Scholl, A. (2008): Sequencing mixed-model assembly lines to minimize part inventory cost. *Operations Research Spectrum* 30, 611–633.
- Boysen, N.; Fliedner, M.; Scholl, A. (2009): Level scheduling under storage constraints. *International Journal of Production Research* 47, 2669–2684.
- Boysen, N.; Fliedner, M.; Scholl, A. (2009): Assembly line balancing: Joint precedence graphs under high product variety. *IIE Transactions* 41, 183–193.

- Emde, S.; Fliedner, M.; Boysen, N. (2009): Optimally loading clocked tow trains for JIT-supply of mixed-model assembly lines. Jena Research Papers in Business and Economics (JBE) 10/2009, FSU Jena, Germany.
- Emde, S.; Boysen, N. (2010): Optimally routing and scheduling tow trains for JIT-supply of mixed-model assembly lines. Jena Research Papers in Business and Economics 08/2010, School of Economics and Business Administration, Friedrich-Schiller-University Jena.
- Golz, J.; Gujjula, R.; Günther, H.-O. (2010): Part feeding at high-variant mixed-model assembly lines. In: Proceedings of the 17th International Annual EurOMA Conference, Porto.
- Hodgson, T.J.; Wang, D. (1991): Optimal hybrid push/pull control strategies for a parallel multistage system: part I. International Journal of Production Research 29, 1279–1287.
- Holweg, M. (2007): The genealogy of lean production. Journal of Operations Management 25, 420–437.
- Johnson, P.F.; Leenders, M.R. (2004): Implementing organizational change in supply towards decentralization. Journal of Purchasing & Supply Management 10, 191–200.
- Klose, A.; Drexl, A. (2005): Facility location models for distribution system design. European Journal of Operational Research 162, 4–29.
- Krarup, J.; Pisinger, D.; Plastria, F. (2002): Discrete location problems with push-pull objectives. Discrete Applied Mathematics 123, 363–378.
- Monden, Y. (1998): Toyota Production System: An integrated approach to just-in-time, third ed. Engineering & Management Press, Norcross, GA.
- Owen, S.H.; Daskin, M.S. (1998): Strategic facility location: A review. European Journal of Operational Research 111, 423–447.
- Rees, L.P.; Huang, P.Y.; Taylor III, B.W. (1989): A comparative analysis of an MRP lot-for-lot system and a Kanban system for a multistage production operation. International Journal of Production Research 27, 1427–1443.
- ReVelle, C.S.; Eiselt, H.A. (2005): Location analysis: A synthesis and survey. European Journal of Operational Research 165, 1–19.
- Saigal, R. (1968): A constrained shortest route problem. Operations Research 16, 205–209.
- de Souza, M.C.; de Carvalho, C.R.V.; Brizon, W.B. (2008): Packing items to feed assembly lines. European Journal of Operational Research 184, 480–489

- Spencer, M. (1995): Production planning in a MRP/JIT repetitive manufacturing environment. *Production Planning & Control* 6, 176–184.
- Vaidyanathan, B.S.; Matson, J.O.; Miller, D.M.; Matson, J.E. (1999): A capacitated vehicle routing problem for just-in- time delivery. *IIE Transactions* 31, 1083–1092.
- Valentinia, G.; Zavanella, L. (2003): The consignment stock of inventories: industrial case and performance analysis. *International Journal of Production Economics* 81–82, 215–224.
- Vatalaro, J.; Taylor, R. (2005): Implementing a Mixed Model Kanban System: The Lean Replenishment Technique for Pull Production. Productivity Press, Portland, OR.
- Wanke, P.F.; Zinn, W. (2004): Strategic logistics decision making. *Interantional Journal of Physical Distribution & Logistics Management* 34, 466–478.